Selbstmanagement in heterogenen Netzen

Claudia Baltes
TU Kaiserslautern
Regionales Hochschulrechenzentrum Kaiserslautern (RHRK)
Paul-Ehrlich-Straße, 67663 Kaiserslautern
baltes@rhrk.uni-kl.de

Bernd Reuther
Fundamental Generic Networking GmbH (FGN)
Gottlieb-Daimler-Straße, 67663 Kaiserslautern
reuther@fg-networking.de

Abstract: Netzwerkadministratoren müssen häufig mit manuellen Eingriffen auf Ereignisse im Netz reagieren. Auf diese Weise kann jedoch keine große Zahl von Ereignissen ausreichend schnell bearbeitet werden. Zudem führen manuelle Eingriffe, insbesondere unter Zeitdruck, immer wieder zu Fehlern. Diese Arbeit stellt eine grundlegende Architektur zur automatischen Reaktion auf unterschiedliche Ereignisse vor. Durch die Berücksichtigung der Netzwerktopologie und gerätespezifischer Eigenschaften ist auch ein Einsatz in heterogenen Umgebungen möglich. Es wird gezeigt, wie die Architektur für ein selbstregulierendes System zur Eindämmung des Netzwerk-Missbrauchs genutzt werden kann. Dabei werden Hinweise auf den Missbrauch als Ereignisse betrachtet, auf die mit der Einschränkung des Netzwerkzugangs bis hin zur vollständigen Trennung eines Nutzers vom Netz reagiert wird.

1 Einleitung

Durch die Einführung neuer Technologien und Integration von VoIP oder anderen interessanten Anwendungen werden Computernetze immer komplexer. Gleichzeitig steigen die Anforderungen an Hochverfügbarkeit, Sicherheit und Mechanismen wie Quality of Service.

Dadurch wird auch die Administration dieser Netze immer aufwendiger. Ohne die entsprechende Unterstützung durch Netzwerkmanagement-Tools werden Computernetze in Zukunft nicht mehr zu warten sein. Mehr und mehr werden diese Tools nicht nur unterstützend arbeiten, sondern selbstständig in die Netzwerkkonfiguration eingreifen.

Sicherheitsbedrohungen wie Viren, Würmer und Trojaner stellen einen Großteil der Probleme in heutigen Netzen dar. Aber auch anderes nicht konformes Verhalten wie das Nutzen von Peer-To-Peer-Protokollen zum Austausch von illegalen Inhalten muss unterbunden werden. Unser Ansatz definiert eine grundlegende Architektur, in dessen Vordergrund ein Regulierungssystem steht, das Sperraufträge entgegen nimmt und automatisch den betroffenen Teilnehmer vom Netz trennt.

Bestehende Lösungen für diese Aufgabe sind bisher sehr spezifisch. Verschiedene Hersteller von Netzwerkkomponenten bieten Intrusion Prevention Systeme an, die aber auf die Produkte dieser Hersteller beschränkt sind. Andere Arbeiten auf diesem Gebiet definieren Lösungen, die speziell auf die entsprechende Netzumgebung zugeschnitten sind und oft ein Gateway, auf dem reguliert wird, voraussetzen.

Dieses Paper stellt eine Lösung vor, die herstellerunabhängig ist und so in heterogenen Netzen eingesetzt werden kann. Um die Lösung so flexibel wie möglich zu gestalten, wurde eine Service-orientierte Architektur ausgewählt. Kern dieser Architektur ist das Regulierungssystem, das auf der einen Seite Regulierungsaufträge entgegen nimmt und auf der anderen Seite Sperr- und Benachrichtigungsaufträge erteilt. Durch die entsprechenden Service-Schnittstellen ist es möglich, das System jederzeit durch notwendige Komponenten zu ergänzen und so an die entsprechende Umgebung und Anforderungen anzupassen. Bestehende, bisher isolierte Tools, die an der TU Kaiserslautern schon vorher zum Einsatz kamen, wurden in diese Gesamtlösung integriert.

2 Architektur

2.1 Überblick

Abbildung 1 stellt die verschiedenen Komponenten der Architektur dar und zeigt das Zusammenspiel des Regulierungssystems mit den übrigen Bausteinen.

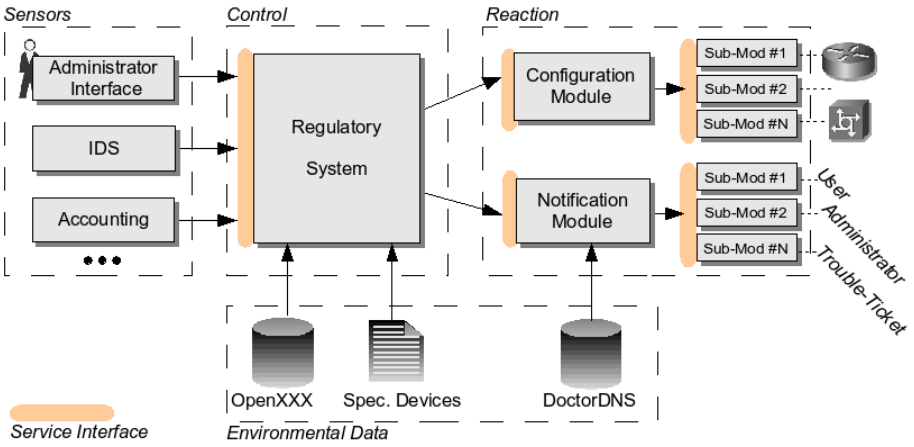

Abbildung 1: Architektur - Komponenten

Verschiedene Sensoren überwachen das Netz und erteilen dem Regulierungssystem bei Bedarf den Auftrag, einen Teilnehmer vom Netz zu trennen. Als Sensoren können zum Beispiel ein IDS oder ein Quota-System eingesetzt werden. Das Regulierungssystem ent-

scheidet, in welcher Art und Weise der Teilnehmer vom Netz getrennt werden soll. Dies kann unter anderem durch Deaktivieren eines Ports, Aktivieren eines Paketfilters, aber auch durch den Einsatz von QoS-Mechanismen wie Rate-Limiting geschehen. Verschiedene Randbedingungen beeinflussen dabei die Entscheidung, an welcher Stelle im Netz gesperrt wird. Der Sensor übergibt mit dem Regulierungsauftrag einen Prioritätswert und bei Wunsch auch eine QoS-Klasse, außerdem muss die Entscheidung natürlich von dem Funktionsumfang der eingesetzten Netzwerkkomponenten abhängen. So sollte ein Rechner, der von einem Virus befallen ist, so nah wie möglich am physikalischen Port gesperrt werden, deswegen sollte der Auftrag mit entsprechend hohem Prioritätswert vom Sensor übergeben werden. Auf der anderen Seite reicht es für einen Rechner, der das Quotalimit für externen Traffic erreicht hat, aus, ihm durch einen Paketfilter den Zugriff aufs Internet zu verwehren.

Ist ein Regulierungsauftrag eingetroffen, wird der Teilnehmer zunächst auf dem Netz lokalisiert. Danach muss die Entscheidung getroffen werden, wie reguliert wird. Dabei spielen Randbedingungen wie Funktionalitäten der Netzwerkkomponenten und Sonderfälle (zum Beispiel das Sperren eines Users auf einem Authentifizierungs-Server) eine Rolle. Der Algorithmus sieht vor, dass solche Randbedingungen aus Spezifikations-Dateien ausgelesen und berücksichtigt werden. Zur Durchführung wird ein Auftrag an die passenden Konfigurations- und Benachrichtigungs-Module gegeben, um den Teilnehmer zu sperren und die zugehörigen Administratoren zu benachrichtigen.

2.2 Sensoren

In der momentanen Testinstallation werden drei verschiedene Sensoren genutzt: Das IDS System Snort [1], ein Accountingsystem basierend auf NetFlow-Daten und natürlich ein Administrator Interface. Das Accountingsystem wird genutzt, um für die Teilnehmer aus den Studentenwohnheimen eine Quota zu realisieren.

Die SOAP Schnittstelle [2],[3] ermöglicht den verschiedenen Sensoren, das Regulierungssystem zu nutzen. Als Parameter muss primär die IP Adresse des zu regulierenden Hosts übergeben werden. Zusätzlich wird die Info benötigt, ob es sich bei dem Regulierungsauftrag um eine Sperrung, Modifizierung oder Freischaltung handelt. Zusätzlich wurde die Option berücksichtigt, eine Regulierung mittels Quality of Service durchzuführen. In beiden Fällen (Regulierung mit oder ohne QoS) wird eine Dringlichkeit für den Sperrauftrag mit übergeben. Dieser Wert zwischen 0 und 255 wird dann vom Regulierungssystem interpretiert. Tabelle 1 beschreibt die momentane Bedeutung der Dringlichkeitsklassen für Regulierung ohne Quality of Service.

2.3 Regulierungssystem

Das Regulierungssystem ist das Kernstück dieser Arbeit. Es nimmt die Regulierungsaufträge der Sensoren entgegen und ermittelt dann, wie dieser Regulierungsauftrag zu ver-

Class	Type of Regulation
0	Disable switch port
1	MAC filter at the switch port
2	IP filter at the first hop router
3	IP filter at the second hop router
...	...
255	IP filter at the administrative boundary

Tabelle 1: Interpretation der Dringlichkeitsklassen

wirklichen ist. Dazu werden Umgebungsdaten und Randbedingungen von anderen Komponenten benötigt und evaluiert. Zum Abschluss werden Konfigurations- und Benachrichtigungsaufträge an die entsprechenden Module geschickt. Dies kann falls erwünscht erst nach einer Bestätigung durch den Administrator erfolgen.

2.3.1 Notwendige Topologieinformationen

Damit das System fehlerfrei arbeiten kann, werden diverse Topologieinformationen benötigt, wie Informationen über Router, Switches, Verbindungen, Forwardingtabellen, Uplinks, usw. Besonders Uplinks haben eine spezielle Bedeutung, da verhindert werden muss, dass sie aus Versehen abgeschaltet werden können.

Betrachtet man die notwendigen Informationen gemäß ISO/OSI-Referenzmodell, so werden auf Layer 2 vor allen Dingen die MAC-Adress-Tabellen benötigt. Mit ihnen kann ermittelt werden, an welchem Port ein Teilnehmer angeschlossen ist. Des Weiteren können sie auch genutzt werden, um zu ermitteln, ob ein Port Access- oder Uplink-Port ist. Auch die Anzahl der hinter einem Access-Port angeschlossenen MAC-Adressen wird im späteren Algorithmus von Bedeutung sein.

Auf Layer 3 werden Routingtabellen und ARP-Informationen benötigt. Anhand der ARP-Tabellen erfolgt die Zuordnung IP-Adresse zu MAC-Adresse, um so den Teilnehmer zu lokalisieren. Mit den Routingtabellen kann der Pfad vom zu regulierenden Host ins Internet ermittelt werden, um so die Regulierung möglichst nahe am Teilnehmer zu realisieren.

All diese Informationen kommen vom Netzwerkmanagement-Tool OpenXXX [4], das von der Firma FGN in Zusammenarbeit mit dem Rechenzentrum der TU Kaiserslautern entwickelt wurde.

In Spezialfällen reicht es eventuell nicht aus, nur den Port oder das Routerinterface zu ermitteln und den Teilnehmer dort zu sperren, sondern Konfigurationen auf Authentifizierungs-, DHCP- oder Proxy-Servern müssen auch angepasst werden. Informationen über diese speziellen Geräte kommen aus einem gesonderten Konfigurationsfile.

2.3.2 Mögliche Regulierungen

Ziel ist es, einen Teilnehmer vom Netz zu trennen. Dies kann auf verschiedene Art und Weise geschehen und ist von den Randbedingungen abhängig.

Am einfachsten ist es, den Switchport, an dem der Teilnehmer angeschlossen ist, zu deaktivieren.

Aber vielleicht muss oder will man eine weniger restriktive Methode nutzen und arbeitet statt dessen mit Paketfiltern. Ein Grund könnte sein, dass hinter dem Switchport mehrere MAC-Adressen vorhanden sind. Dies kommt recht häufig vor, da die strukturierte Verkabelung oft nicht ausreicht und so nicht genügend Datenports in den Räumen zur Verfügung stehen.

Eine Möglichkeit ist es, einen Filter auf Layer 2 zu nutzen und so die entsprechende MAC-Adresse zu filtern.

Eine Alternative ist das Filtern auf Layer 3, dies kann je nach Hardware auch schon auf dem Switchport geschehen, wird aber typischerweise eher auf einem Router realisiert. Möglich ist dabei jeder Router auf dem Pfad von zu regulierendem Host Richtung Internet, das Filtern sollte aber im Allgemeinen so nah wie möglich am Host geschehen.

Eine Variante zum Sperren ist es, eine QoS Klasse anzuwenden, in der zum Beispiel ein Rate Limiting für den zu regulierenden Teilnehmer definiert ist.

Besonderes Augenmerk sollte auf verschiedene Sonderfälle gelegt werden, in denen es sinnvoll ist, (zusätzlich) die Konfiguration eines Servers zu modifizieren. Ein kleines Beispiel zur Illustration: Soll eine auffällige IP-Adresse gesperrt werden, so kann zum Beispiel der zugehörige Port gefunden und gesperrt werden. Um zu verhindern, dass der Rechner jetzt aber an der Nachbardose genutzt wird, könnte man zusätzlich die MAC-Adresse auf dem DHCP-Server sperren. So kann entsprechend auch ein User auf einem Authentifizierungsserver (wie Radius) oder eine IP-Adresse auf einem Proxy-Server gesperrt werden. Letzteres verhindert, dass eventuell ein Paketfilter durch das Nutzen des Proxy-Servers umgangen werden kann.

2.3.3 Regulierungsalgorithmus

Das Regulierungssystem bestimmt automatisch, welche der möglichen Regulierungen für den eingegangenen Regulierungsauftrag durchgeführt werden soll. Dies geschieht gemäß folgendem Algorithmus (vergleiche Abbildung 2):

Als Parameter übergeben die Sensoren im Regulierungsauftrag die IP-Adresse des zu regulierenden Hosts, die Dringlichkeitsklasse und die Information, ob mit oder ohne QoS reguliert werden soll. Allerdings wird als Dringlichkeit ein Intervall übergeben, um so dem Regulierungssystem Spielraum zu geben, die optimale Regulierungsvariante zu ermitteln.

Als erstes muss überprüft werden, ob der Host schon reguliert ist. Falls ja, stellt sich die Frage, ob die neue Regulierung eine höhere Dringlichkeitsstufe hat als die alte. Falls ja, muss sie natürlich ausgeführt werden, falls nein, muss diese neue Regulierung gespeichert werden, so dass sie ausgeführt wird, sobald die alte aufgehoben ist. Wird sie gespeichert,

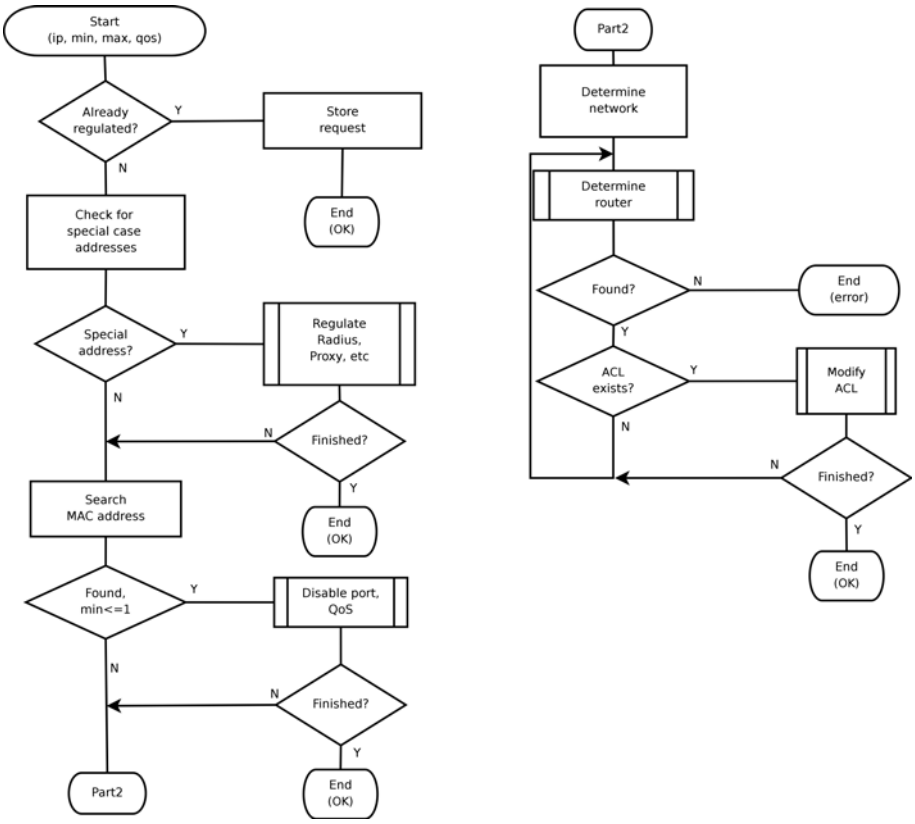

Abbildung 2: Regulierungssystem - Algorithmus

so ist der Algorithmus damit beendet.

Kommt es zu einer Regulierung, wird erst einmal überprüft, ob die IP-Adresse im Bereich der Sonderfälle liegt. Falls ja, muss eine Konfigurationsänderung auf dem entsprechenden Server initiiert werden. Danach stellt sich die Frage, ob die Regulierung damit schon abgeschlossen ist oder nicht. Falls ja, ist der Algorithmus beendet, falls nein, muss der Host im Netz lokalisiert werden.

Dazu wird über die MAC-Adresse der Port ermittelt, an dem der Host angeschlossen ist. Das weitere Vorgehen hängt natürlich von der Dringlichkeitsklasse ab. Ist der untere Wert des Dringlichkeitsintervalls kleiner oder gleich 1, so wird der Port wenn möglich abgeschaltet. Sollten mehrere MAC-Adressen hinter dem Port sein, wird stattdessen ein MAC-Filter angewandt.

Ist beides nicht möglich oder nicht erwünscht, also die untere Dringlichkeitsgrenze größer 1, so muss (sofern die obere Dringlichkeitsgrenze das noch erlaubt) stattdessen mittels IP-Filter reguliert werden. Dazu ist es nötig, den Router zu ermitteln, der als Default Gate-

way für das Netz fungiert. Dies erfolgt über Auswertung der Routingtabellen. Auf diesem Router darf allerdings nur reguliert werden, falls eine vordefinierte Accessliste existiert, die dann modifiziert wird. Ist dies nicht der Fall, so muss der nächste Router gesucht werden, um zu überprüfen, ob hier mittels ACL reguliert werden kann. Diese Schleife wird fortgesetzt, bis entweder reguliert wird, die obere Dringlichkeitsklasse erreicht ist oder aber kein weiterer Router mehr im Pfad zwischen Host und 'Internetgateway' vorhanden ist. Dann wird der Algorithmus mit entsprechender Fehlermeldung abschließen.

In jedem Fall wird natürlich eine Rückmeldung an den Sensor gegeben, ob der Regulierungsauftrag durchgeführt wurde oder nicht.

2.4 Konfiguration und Benachrichtigung

Nachdem das Regulierungssystem entschieden hat, auf welche Art und Weise die Regulierung durchgeführt werden soll, müssen dementsprechende Konfigurations- und Benachrichtigungsaufträge verschickt werden.

Da die unterstützten Managementprotokolle und Konfigurationsbefehle stark von Hersteller und Hardware abhängen, ist es nötig, für verschiedene Aufgaben und Hersteller die passenden Konfigurationssubmodule bereitzustellen.

Soweit möglich wurde SNMP als Managementprotokoll genutzt. Besteht der Konfigurationsauftrag zum Beispiel darin, einen Port zu deaktivieren, so ist das problemlos per Standard-SNMP-MIB möglich.

Anders sieht es zum Beispiel aus, wenn die MAC-Adresse auf Portebene gefiltert werden soll. Die TU Kaiserslautern hat, wie viele Universitäten, ein recht heterogenes Netz, und besonders im Access Layer findet man Switches jeglicher Couleur. Um nun einen MAC-Filter zu realisieren, nutzt Cisco Systems eine Layer 2 Accessliste, Enterasys Networks eine Policy, Extreme Networks einen Blackhole FDB Entry und die alten Komponenten von Fore Systems unterstützen einen solchen Filter nicht einmal.

Die modulare Struktur unserer Lösung bietet daher die Möglichkeit, bei Bedarf jederzeit Konfigurations(sub)module zu ergänzen.

Ein weiterer wesentlicher Bestandteil ist natürlich das Benachrichtigungsmodul. Als Minimum ist hier eine Meldung im Logging des Systems und eine Benachrichtigung des Administrators (zum Beispiel über Email) anzusehen. Mehr ist im Moment noch nicht realisiert.

Nächster Schritt wird die Integration ins Troubleticket-System des Rechenzentrums sein.

Allerdings bieten sich einige weitere interessante Ideen an. Ist der User bekannt (zum Beispiel dadurch, dass IEEE 802.1X genutzt wird), so kann man natürlich eine Email an den Benutzer schicken lassen.

An der TU Kaiserslautern gibt es für Netzwerkfragen der Fachbereiche so genannte Fachbereichsadministratoren, des Weiteren haben die einzelnen Arbeitsgruppen oft zusätzliche Ansprechpartner. Auch hier wäre es interessant, diese lokalen Administratoren automa-

tisch über den Regulierungsfall zu benachrichtigen. Dazu soll das an der TU Kaiserslautern genutzte DNS-, DHCP- und Asset-Management-System DoctorDNS [5] eingesetzt werden, in dem neben DNS und DHCP Daten auch Lokation und Ansprechpartner der Rechner gepflegt werden.

Eine weitere Möglichkeit bietet sich, wenn in DoctorDNS auch Telefonnummern verwaltet werden. In diesem Fall wäre ein automatisierter Anruf denkbar. Auch IP Telefone könnten eine interessante Variante bieten: Im IP Telefon ist ein integrierter Switch, wird dieser zum Sperren genutzt, so ist der Ort der Sperrung mittels Telefonanruf oder Nachricht auf dem Telefon direkt zu erreichen und kann so informiert werden.

3 Zusammenfassung und Ausblick

Die Service-orientierte Architektur unseres Ansatzes bildet die Basis für das Selbstmanagement in heterogenen Netzen. Durch die offenen Schnittstellen ist es einfach, das System um neue Sensoren oder Aktuatoren zu erweitern. Die Prototyp Installation ist sehr viel versprechend und soll in absehbarer Zeit im Produktivnetz eingesetzt werden.

Die Sonderfälle sind bisher nicht alle berücksichtigt, da es zum Beispiel noch Defizite im Konzept der Authentifizierungs-Server der TU Kaiserslautern gab. Des Weiteren stellt sich die Frage, ob der Algorithmus zur Ermittlung des Pfades von zu regulierendem Host Richtung Internet auch mit asymmetrischem Routing (das man nicht nutzen sollte, aber vorkommen kann) funktionieren wird. Ein wichtiger Punkt, der noch fehlt, ist die Überwachung der Konfigurationsdateien der Netzwerkkomponenten. Zur Zeit fängt das System nicht ab, wenn ein Administrator einen Port manuell wieder aktiviert, statt dies über einen Regulierungsauftrag anzustoßen.

Zusammenfassend bleibt zu sagen, dass das System, entstanden aus einer Diplomarbeit [6], eine hervorragende Unterstützung für die tägliche Arbeit eines Netzwerkadministrators bildet. Hier näher vorgestellt wurde die Komponente des Regulierungssystems zur automatischen Regulierung von nicht konformen Netzteilnehmern. Die Architektur bildet aber die Grundlage für eine Reihe von weiteren Arbeiten zur Integration und Automatisierung von Tools im Netzwerkmanagement-Bereich.

Literatur

[1] SNORT. „The Open Source Network Intrusion Detection System" [Online] `http://www.snort.org/`.

[2] M. Gudgin, M. Hadley, J. J. Moreau, H. Frystyk Nielsen „SOAP version 1.2" World Wide Web Consortium, Working Draft WD-soap12-20010709, July 2001.

[3] M. Gudgin, M. Hadley, J. J. Moreau, H. Frystyk Nielsen „SOAP version 1.2 part 1: Messaging framework" World Wide Web Consortium, Recommendation REC-soap12-part1-20030624, June 2003.

[4] fgn GmbH. OpenXXX. [Online] http://www.openxxx.de/.

[5] fgn GmbH. DoctorDNS. [Online] http://www.doctordns.org/.

[6] Patrick Koppen. Diplomarbeit. Eine servicebasierte Architektur für selbstorganisierende Netzwerke.